Integrated Control Methodologies for Road Vehicles

TIM GORDON, MARK HOWELL
AND FELIPE BRANDAO *

SUMMARY

This paper considers the scope, methodologies and architectures for the design and development of interacting control systems in road vehicles. The increasing use of electronic controls leads inevitably to an increase in overall system complexity. Given the time and economic constraints of the modern automotive industry, it is not feasible to synthesise and validate the full set of vehicle controls in the form of a unified and centralized controller. On the other hand a fully decentralized approach to control system development and operation will induce performance limitations from un-modelled or unexpected interactions; at worst, such interactions can cause instability and loss of function. There is now increasing pressure to achieve control coordination whilst maintaining a modular approach to the overall system design. With this in mind, the paper provides a framework to review current practice in integrated vehicle control, assesses recent developments in control integration methodologies that are most relevant to the vehicle application, and formulates an enhanced multi-layer architecture that includes explicit coordination functionality. Overall emphasis is placed on the role of control system architecture, the resulting flow of control information and the implications for control system design. An example from handling dynamics is presented, demonstrating the viability of new and flexible approaches. In conclusion a number of outstanding research problems are highlighted.

1. INTRODUCTION

Any control system comprises hardware such as sensors, actuators, communications links, power electronics, switches and micro-processors. Such hardware is becoming increasingly common on modern motor vehicles, with 30-50 microprocessors not uncommon on a single vehicle [1]. Hardware is typically distributed around the vehicle, but of course the physical location tells us relatively little about the design of the functional control structure, which is the main topic of this paper.

In the past it has been common for separate vehicle functions to be controlled 'independently', or rather in parallel. This means that control hardware can be grouped into discrete subsets, with sensor information and control demands operating in parallel processes and with no possible ambiguity or conflict over the responses demanded of the actuators. Of course interactions may occur, when for example brake and rear-steer

* Address for correspondence: Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, Leicestershire, LE11 3TU, England. Tel +44 1509 227201, email t.j.gordon@lboro.ac.uk

actuators control the same (yaw) degree of freedom of the vehicle. We shall refer to this as a parallel or decentralised control architecture; the term *architecture* will be taken to mean an abstraction of the pattern (or topology) of how sensor information and control commands interact between various control sub-systems and components. Indeed we shall seek to resolve the term architecture into more detail as it becomes clearer that the control structure operates at a number of levels; for example an important aspect is the pattern of sub-system functionality (the assignment of control objectives to sub-systems) designed into the vehicle control system. This makes the overall control architecture a deeper concept that simply the physical layout of control components, or even the causal relationship between distributed control processes.

The parallel vehicle control architecture has largely arisen by default, as different controlled sub-systems are developed and manufactured by independent supplier companies, or by different groups within a vehicle manufacturer. But there is a very limited number of relevant degrees of freedom in a typical car: for motion control there are six rigid-body modes, and even adding the major system states of engine, transmission, wheel spin and wheel-hop, the number is still only around 20. So as the number of control functions and actuators increases, it is inevitable that interactions and performance conflicts arise within any highly parallel architecture. For example, the use of single-wheel braking to reduce oversteer or understeer will certainly conflict with the requirement for traction under forward acceleration. The unsatisfactory nature of this has been recognised for some time (e.g. [2-5]) and yet the overall issue remains [6]. The foremost concerns are to do with reducing complexity, improving performance and removing unnecessary and costly duplication of hardware [6]. Coelingh [6] presents an illustration similar to that in Fig. 1, illustrating the way that a 'Complete Vehicle Control' structure might limit the growth in overall system complexity as the number of actuators and sensors increases. As a global structure is imposed, complexity is reduced, at least in the physical structure of the control system; however as is also suggested by Fig. 1. the complexity may simply be masked, preserved intact within the software.

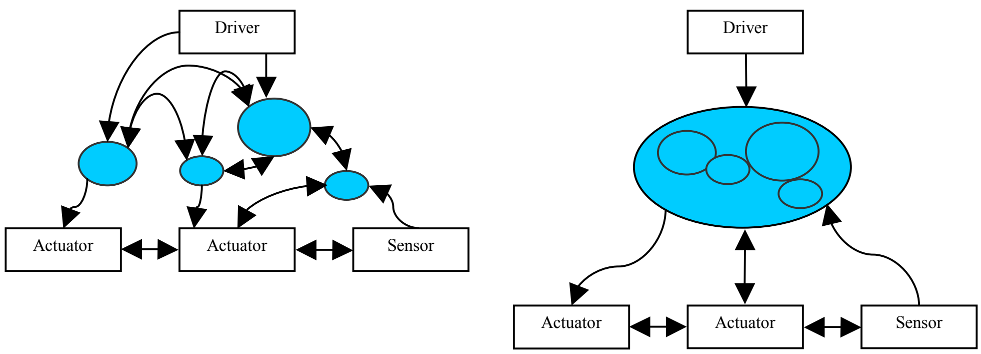

Fig. 1. Unstructured vs. structured control architectures

Overall, in the development of an Integrated Vehicle Control System (IVCS), the aim is to combine and supervise all controllable subsystems affecting vehicle dynamic response [2-4,7-9] to: improve multiple-objective performance from available actuators, by removing the implicit constraint of a one-to-one link between control objectives and particular sets of sensors and actuators

- reduce or contain complexity at the number of control components increases
- improve safety and comfort
- reduce or contain system costs, by avoiding unnecessary duplication of components and by information sharing between sensors.
- improve flexibility
    - permitting a modular and distributed design process
    - moving towards 'plug-and-play' extensibility
    - incorporating diagnostics and condition monitoring
    - improving system reliability
    - allowing fault-tolerant and failure management capability

There is a close link between flexible plug-and-play design, and fault-tolerant control system operation; given the availability of suitable diagnostics, condition monitoring, and adaptability in the control algorithms, should a sensor or actuator fail, the failed component may be de-activated and reconfiguration take place to minimise degradation, or at least provide a safe and predictable failure mode for the vehicle. In this way, it should be possible to generate redundancy in a distributed way at the system level, which is cheaper and more effective than simply duplicating key components. One particular algorithmic device to achieve such behaviour, based on formal optimal control, has been presented by one of the authors [10] but there appears to be scope for significant future developments in this area – see Section 6 below.

Further aspects of IVCS flexibility are described in the discussion of integrated control for helicopter dynamics by Wills et.al. [11]. In particular, this paper highlights the importance of *interoperability* and *openness*. The former is a relatively low-level matter dealing with the need for effective communication between different microprocessors (e.g. using a CAN bus). This is an essential aspect of any modular approach to system design - rigorous interfacing standards must be applied for anything to actually work! On the other hand, openness refers to the need for software, algorithms and their resources to be sufficiently open to allow for systematic integration. This is a key point, especially for automotive applications, since in the foreseeable future, proprietary control algorithms will not be freely made available; but this need not prevent an open architecture operating between IVCS modules.

Flexibility also encompasses the need for a modular approach to design and manufacture. Vehicles are not designed and built in isolation; if a manufacturer sells five different vehicles, each with five optional control sub-systems that may or not be implemented, and each employs alternative hardware from two different suppliers, there are $5 \times 3^5 = 1215$ possible combinations, each potentially requiring a unique variant of an overall vehicle control algorithm; while these numbers are hypothetical, the problem is not. If the controlling software is not modular, design and development becomes unmanageable.

These issues provide the motivation for this paper: to clarify the conceptual basis and future challenges of integrated vehicle control. To achieve this, we review the current state of the art in Integrated Vehicle Control Systems, with a critical emphasis on functional control structures and operational architectures. The focus is on the area of vehicle motion control, with particular emphasis on chassis systems. The aim is to highlight and demonstrate the major options available, and provide a degree of synthesis of ideas – hopefully to move towards a clear and consistent approach to identify suitable IVCS architectures, that not only encompass reported work to date, but also suggests new possibilities for the future.

As a self-imposed limitation, the important areas of integrated sensor fusion, state estimation and information management will not be considered in any detail; it will mostly be assumed that the various vehicle or sub-system controllers have direct access to sufficient sensor or state information for control decisions to be made. This stretches the normal use of the well-known 'Separation Principle' whereby control and estimation functions may be developed independently. It is recognised however that issues of fault tolerance and overall system complexity cannot be fully addressed without taking account of the sensor and information aspects. The concept of the vehicle as a subsystem interacting with 'intelligent' external roadway and traffic systems is also not considered in any detail (see the review in [12] for more details). However it is clear that IVCS 'expandability' is an important issue when such external interactions are introduced. For example [13] the external control of vehicle speed can be beneficial in reducing congestion, and the automation of speed control for traffic management, platooning etc. becomes technically and economically very feasible once sufficient vehicles are fitted with drive-by-wire systems; the global control architecture must be compatible with that of the individual vehicles, and the various (local and global) fault/failure-mode behaviours may become a critically important implementation issue for the future.

In the next section we review the state of the art of IVCS application areas, in particular to understand where vehicle control integration offers significant performance benefits, and also which general control algorithms have been used to support such integration. In Section 3 some basic architectures are considered, as are the levels at which these operate. Section 4 deals with multi-layered 'behaviour based' architectures in more detail, while Section 5 looks more deeply at the coordination mechanisms available for control integration. This is followed by an illustrative simulation study for handling control, using layered design and combined fuzzy/subsumption coordination. Section 6 looks at the scope for extending the capabilities of coordinators so that they are more systematically linked to the performance criteria of the associated sub-system controllers. Finally Section 7 brings together the main conclusions of the paper, and highlights a number of areas for future research.

2. INTEGRATED VEHICLE CONTROL APPLICATIONS

The fundamental objective of vehicle motion control is to direct the vehicle, according to driver demands, in the longitudinal and lateral directions whilst maintaining acceptable variations in bounce, pitch, roll and vehicle slip angle. Although body roll is intimately linked with handling performance, it is largely a 'means to an end' (driver feedback, bump-steer etc.) rather than a primary goal of vehicle dynamics. The aim is to provide safe, predictable and responsive motion control, whilst maintaining acceptable occupant ride comfort. NVH, energy consumption and similar issues are clearly important, but again play a secondary role in the current discussion. In principle, suspension control is also secondary - the driver has no direct influence over this sub-system; however, in practice, its direct influence on vertical tyre loads and the vertical, pitch and roll degrees of freedom clearly give it a very important role, and it is normally treated on an equal footing to braking and yaw-moment control for example.

Wallentowitz [4] considered cross-compatibility of sensors and processor operations in current and potential future vehicles. In this broad review, the case has been made for a number of integration improvements for chassis-specific functions such as ABS, stability and traction control under acceleration, active rear-axle kinematics, all-wheel drive and lateral dynamic stability, as well as active ride dynamics. The paper makes the case based largely on sensor-sharing and sensor redundancy issues, but also describes the significant problem of 'ownership and distribution of IPR' mentioned in Section 1.

In another paper on the 'basic principles' of integrated control, Kiencke [5] gives a general overview on how the introduction of Control Area Networking (CAN) provides convenient scope for the integration of previously independent systems. The work reinforces the above idea that it is fundamentally the small number of vehicle rigid-body degrees of freedom compared to available actuators that drives the performance advantages of IVCS; Kiencke focuses on the combined use of brakes and rear-steer to augment the driver's front-steer input in controlling the yaw (and implicitly sideslip) degree of freedom. A sliding mode algorithm, which involves the real-time estimation of tyre friction, is described for the integrated control.

To better understand the overall potential performance benefits of integrated motion control of the vehicle, the basic task can be represented via a 'g-g diagram' of vehicle mass centre lateral and longitudinal accelerations. The driver uses accelerator, brakes and steering to maintain control of the magnitude and direction of the mass-centre velocity vector; changes in this vector are then the g-g diagram accelerations. In terms of the 'ideal function' of motion control, the driver is limited only by the friction constraints of the tyres - plus of course in-plane aerodynamic forces when these become significant at high speeds. The vehicle controls are expected to provide the driver with predictable authority over these accelerations, within the physical constraints of the vehicle 'friction circle', and subject to perceived customer acceptability of the frequency and amplitude dependence of the vehicle responses. Also, given that friction limits change with speed and road surface condition etc., these vehicle control systems are required to provide adequate feedback of such changes - for example via steering torque. The concept in Fig. 2 was presented by Tanaka et al [3] as a schematic to indicate the domain of operation of some typical vehicle control systems, and the areas where system integration are likely to be beneficial. Although the diagram should not be taken too literally, it clearly underlines the conclusion from the Toyota Soarer that integrated control can enlarge and 'smooth out' the dynamic response domain of the vehicle.

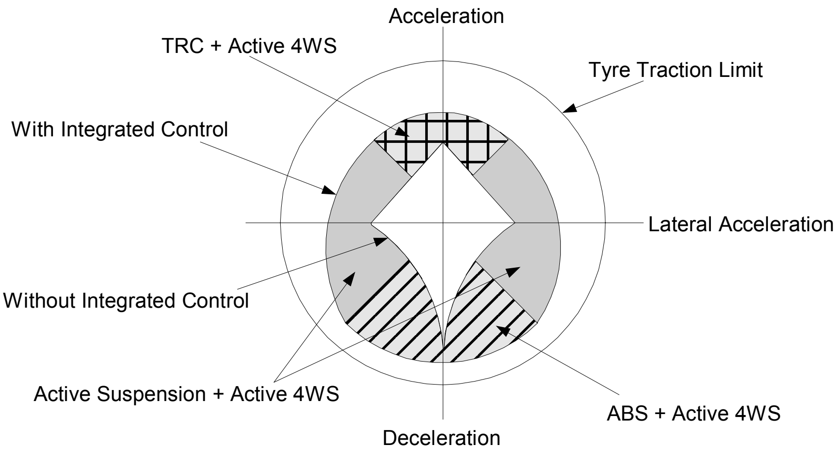

Fig. 2. Integrated Control and the g-g Diagram

The Toyota Soarer employed combined control of hydro-pneumatic suspension, four wheel steer and active ABS / traction control [3,14]. Making use of classical control techniques, with three term (PID) feedback for each subsystem, control integration was based on logical mode detection (for example at preset thresholds of longitudinal and lateral accelerations) to regulate controller gains at a supervisory level. The published results epitomise the degree of control improvement that can be achieved, even without any recourse to any particular formal or multivariable control algorithms.

In a later paper from Toyota, Hirano et al [7] advocated a similar ethos, implementing a four wheel steer/ four wheel drive (4WS/4WD) controller via feedforward and feedback compensators designed using multivariable H_∞ methods; these are suitably inter-related between each subsystem, and adapt for saturating tyre forces. The authors report improved vehicle stability on slippery surfaces and an improved steering response in both simulation and experiment. The experimental vehicle employed Local Area Network (LAN) communications to link a central control unit to a distributed set of five local control units for 4WS (for the rear steer actuator) 4WD (for actuating drive torque distribution), ABS, engine control and electronic throttle. Though only 4WS/4WD were explicitly integrated within the control algorithm, this paper indicates a physical architecture that is relatively fault-tolerant: if there is an interruption or failure of the central controller or the LAN, the local control units can continue to provide a safe basic level of functionality. Note however that this architecture is not explicitly reflected in the controller design methodology.

Given the consistent theme of performance improvement by integrated (or simply multivariable) control, it is worth focussing further attention on algorithms employed, and the extent to which these can support a modular and distributed architecture.

A number of authors have approached the design of integrated vehicle controllers via a full-vehicle reference model. The dynamic influence of various actuators are incorporated via input matrices in whole-vehicle MIMO model, which are typically of low order. These are applied in an off-line control law design, resulting in improved vehicle performance, for example in handling stability, when compared with the use of independent controllers. Examples include model matched-control using a model-inversion [15,16], robust H_2 and H_∞ design methods [17], and nonlinear predictive control [18]. Whilst the these papers are purely based on simulation investigations, there is a consistent (and perhaps obvious) conclusion that a properly designed integrating controller improves handling performance over a wide range of manoeuvres, strongly endorsing the conceptual g-g diagram of Tanaka et al.

Other authors have employed standard techniques such as direct output feedback methods (single-loop analogue or discrete-time compensators) [3,7,19,20], sliding mode control [5,21-23], model reference [16,24], fuzzy logic [25-27] and Artificial Neural Networks [28,29]. Two particularly common formal control methods, both essentially based on linear systems are those of robust H_2 and H_∞ control [7,15,17,30] and optimal control [10,15,31]. These more 'popular' approaches are attractive from an academic standpoint in that they can be easily related to commonly available linear and nonlinear handling dynamic models. The robust algorithms use linear reference models, with uncertainty bounds related to expected errors due to nonlinearity. The optimal control papers typically use a linearised model to design a feedback law, and test this in simulation of a nonlinear model, either directly, or using feedback signals derived by reference to a nonlinear model with certain 'desired' handling characteristics. Both of the more common methods are also attractive for further development, because they offer a clearly 'visible' design technique well-suited to calibration and development via explicit performance indices or frequency weighting functions.

When focussing attention on control algorithms, it is easy to make the mistake of identifying 'integrated control' with 'multivariable' control, hence overlooking the many issues highlighted already in Section 1, especially those relating to control system flexibility. In fact, most of the authors cited in the above fall into this category, the exceptions being references [10,28]. In [10] the emphasis is on modular/distributed control system design, with parallel development of subsystem model-based optimal controls. Synthesis into a coordinated vehicle controller is then centralised, so there are essentially two levels to the control hierarchy. In [28] a model-based control is explicitly introduced to control the six rigid-body degrees of freedom, potentially including driver inputs for longitudinal and lateral control, or – as particularly described in the paper – to integrate the vehicle motion within an automated vehicle highway system. Lower level 'neuro-compensator' controls operate to help ensure the overall vehicle behaves like the reference design model, and hence improve robustness. This two-level hierarchical structure could clearly be developed with further levels of cascaded control. These two approaches, based of specific algorithms, neatly highlight the broad links between algorithmic structure and control architecture for integrated controls; in these particular cases, the first defines a bottom-up distributed design approach, while the second follows a top-down hierarchical structure.

Links between algorithms and supporting architectures are also to be found in related application areas – as in unmanned air vehicles (e.g. [11,32]) and autonomous underwater vehicles (e.g. [30,33]). In references [11,32] a structured hierarchical structure is employed, and in both cases the key algorithmic approach to achieving flexible control is via mode recognition and switching, with baseline control algorithms designed off-line, the details of which are largely irrelevant, but might typically be based on single-loop classical control, or multivariable 'modern' control. In [30] this same approach to mode-switching is proposed, though the baseline algorithms are specifically prescribed as robust multivariable H_∞ control. A limitation with these mode-switching algorithms is that the design stage must be used to pre-empt and predict all likely failure events or changes in operating mode, something that greatly increases the off-line design task and is particularly undesirable in the automotive systems environment. In [33] a quite different approach is taken, with a number of bottom-up behaviour-based approaches being considered. Key features of these approaches will be taken up in the following sections, once the essential elements of vehicle control architectures have been reviewed and refined.

3. VEHICLE CONTROL ARCHITECTURES

The term 'architecture' is widely used [2,6,8,11,12,34] but often loosely so, and with different specific meanings by different authors. Here the intended meaning is as a comprehensive representation of the global IVCS structure, both in its operation and its design. The term therefore encompasses a range of relationships, or 'topologies', connecting various sub-systems or modules. In increasing order of abstraction, there appear to be five potentially distinct levels of topology that comprise the overall control architecture:-

- T1. the *physical layout* of microcontrollers, routers, communication links, sensors, and actuators
- T2. the causal connection and relative authority of *control actions* – incorporating the connectivity of demands, reference signals and any coordination and protection mechanisms
- T3. the connection and flow of information – from sensors, state estimators, control output, condition monitoring and diagnostics

T4. the structure of the *control algorithms and methodologies* that underpin the above, for example that help guarantee stable and fault-tolerant operation of the overall IVCS

T5. the underlying structure of the *control functional design* - as for example in hierarchical or layered approaches where a modular design is used to cascade or subsume control objectives

Again, the term ‘topology’ will be assumed to have a wider scope than is common, not only to include the ‘geometric’ topology of network nodes and links, but also relevant aspect of causality (*A* triggers *B*), ranking (*A* overrides *B*), and dependency (*A* requires *B* in order to function).

Some authors refer to the above *T2*-topology as a ‘functional decomposition’ [6,35] of the integrated control system, emphasising the difference between *T1* and *T2*. However this can easily be confused with *T5*, where functionality is also decomposed into modules in the *design* process. Regarding *T3*, there is relatively little attention given to this in the literature, except in the aspects of fault diagnosis and condition monitoring [36-38].

The topology at one level does not uniquely constrain the topology at other levels. For example, consider the traditional parallel/distributed approach seen in non-integrated vehicle control systems. A parallel structure exists at all levels, except in *T5* where a vehicle manufacturer typically specifies a hierarchy of functional objectives and quantified performance targets. As a move towards ‘control system integration’ the *T1*-topology may be changed from using separate wires for communication, to a networked ‘architecture’ (topology) – as in Fig. 3. Here the shaded bubbles represent microcontrollers associated with specific control functions, and the lines represent physical communication channels. The second *T1*-topology remains compatible with the traditional parallel design, but offers new opportunities at other levels. This simple example underlines the fact that there really are a multiplicity of levels at which control system topologies operate, and there is not a unique association of compatible topologies; it also serves to demonstrate the simplistic nature of using a figure like this as an overall conceptual model for integrated vehicle control!

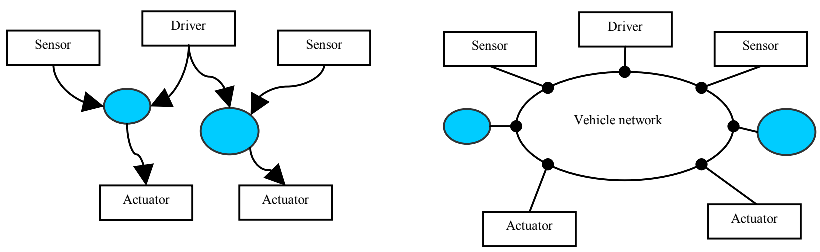

Fig. 3. Parallel versus network implementation of a parallel design topology

Basic issues of the physical topology are also highlighted in the paper by Schmidt et al [36]; by drawing sub-system boundaries in places where the communication traffic is low, the effect of an unavailable link can be minimised. Once again, there is clearly no essential link from this to the supporting control algorithms (*T4*), or functional decomposition in design (*T5*).

Following the self-imposed limitations described in the Introduction, focus will primarily be on the topology levels *T2* and *T5*, though these will be related to the other levels where relevant. Indeed, in the future it may be that *T4* will provide the 'defining' topology: without a proper mathematical formulation at the algorithmic level, it is hard to see how key properties of stability, reliability etc. can be validated for an integrated control system design.

Both *T2* and *T5* may be classified to some extent by the degree of centralisation. The *T5*-topology of a centralised controller may look something like that on the left of Fig. 4. This represents a total vehicle functional control hierarchy, with arrows representing the cascade of control objectives from the vehicle level to supporting control functions. A less centralised approach is represented on the right of the figure, where some more restricted objectives give rise to a number of parallel hierarchies. In all of these cases however the downwards arrows indicate that the lower level control design cannot be carried out until the higher levels have been suitably defined.

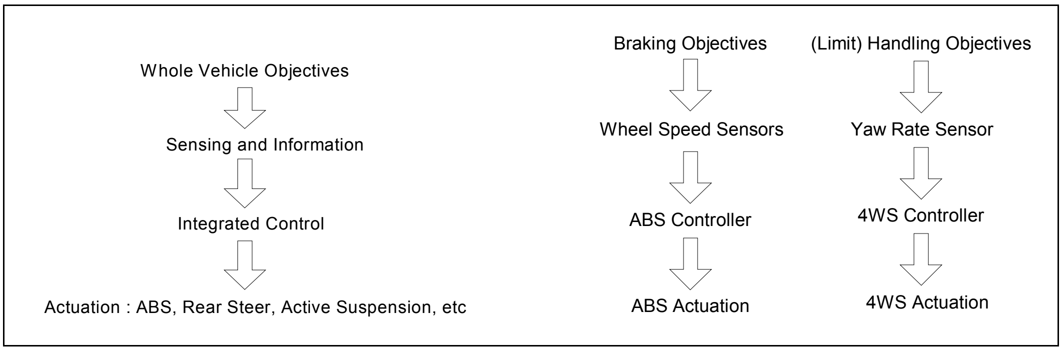

Fig. 4. Single versus parallel functional hierarchies

We now consider three common *T2*-topologies in decreasing order of centralisation, and return to the major role of *T5* in Section 4.

3.1 Centralised Control

Here a central 'total vehicle controller' is responsible for making all control decisions. Therefore, as depicted in Fig. 5, control, sensor and diagnostic information is routed vertically between the two layers of controller and vehicle hardware (actuators and basic mechanical components). The horizontal arrows represent mechanical or other physical interactions directly within the vehicle, the major aspects of which should be modelled and accounted for in the controller design. While the flexibility limitations of such centralised control for vehicles is already clear, there is the potential advantage that stability and performance predictions can be made in a unified manner at the design stage.

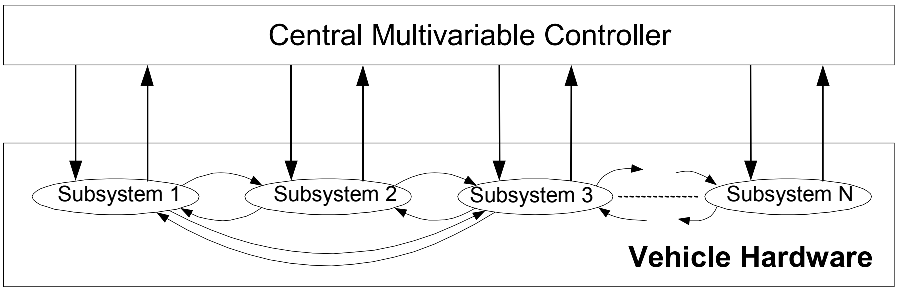

Fig. 5. Centralised control structure

There are many examples in the literature where integrated vehicle control is identified with this architecture, typically linked to the application of a global multivariable control formalism – [7,8,15-18,31,39-41]. When the *T2*-topology is implemented directly into hardware (*T1*-topology) any desired fail-safe redundancy of microcontrollers, power converters etc. increases the number and cost of control components [42]. The option preferred by Kelling and Heck [42], in the context of a brake-by-wire system, is to distribute the brake ECU's to the wheel locations. In the *T4* sense, the control topology remains centralised, but each brake ECU calculates global controls in parallel, and these are communicated between the ECU's and a validation/voting system provides supervision and failure mode capability. In the *T2* (and *T1*) sense the control topology is actually in the form of *supervisory control*.

3.2 Supervisory Control

Supervisory control represents a range of topologies intermediate between fully centralised and fully decentralised, and is established by adding an intermediate layer of local control to the former, or adding a level of supervision to the latter – Fig. 6. Subsystem controllers may then be designed and validated relatively independently. One particular factor is whether the intermediate layer is fully dependent on the supervisory layer – and hence whether effective control is possible in the event of a communication failure between them. For example in the braking system described by Kelling and Heck [42] not only is the latter possible, the (voting-based) supervisory control function is physically distributed – there is no physical central control 'module' at all! On the other hand, in the system described in [6], while some lower level functionality (sensor and sub-system diagnostics) is devolved to the intermediate layer, the essential control task remains centralised, and the loss of communication envisaged could not be tolerated by the IVCS.

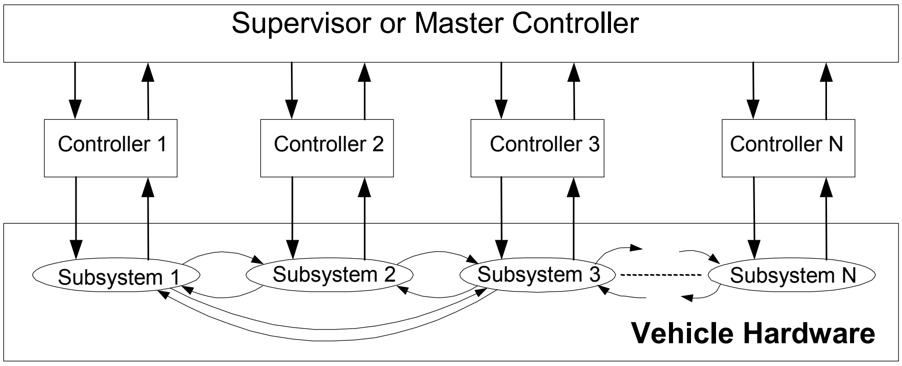

Fig. 6. Supervisory control

The role of the Master Controller may be to supply set-points within a familiar cascade structure [2], or provide mode switching [30] or both [11]. In the 'Open Control Platform' of Wills et al [11], the lowest layer incorporates actuators and low-level control (e.g. set-point tracking) and the supervisory layer works at the 'mission' level, including all sensor input, system condition monitoring, situational awareness etc. The intermediate layer consists of 'traditional' multivariable controllers, e.g. for the (helicopter) flight dynamics control. For this particular topology, the routing of all sensor functions through the highest layer means that again, communications failure to the supervisor (or a major processing failure within it) would cause overall system dysfunction. As mentioned in Section 2, the same general approach is taken by Katebi and Grimble [30] for underwater vehicle control. While it is clear that this type of *T2*-topology supports a basic form of reconfigurable control, doing so in a more general case, in response to non preset scenarios, is a challenging task requiring a degree of 'intelligence' not addressed in these references.

Returning to the control of road vehicles, in the work of Freuchte et al [2], relating to General Motors' "Project Trilby" on integrated vehicle control, the same general three layer *T2*-topology is proposed, though in this case the supervisory layer is further resolved into those aspects that adapt to changing situations within the vehicle, and those that adapt to the driver; a third sub-layer relating to the external highway environment would also seem a natural extension to these ideas. This work also provides a reminder of the essential role of human factors issues in integrated vehicle control, though in [2] much of the description is aspirational in nature.

From the above, it is clear that the three-layer supervisory (*T2*) control topology has a number of attractive features:

- it can potentially operate even if the supervisor malfunctions or high-level communications fail
- it can be extended to a multi-layered hierarchical structure (see Section 4)

- it generates a requirement for consistent interfaces and consistent design function concepts that is reassuring to a conservative industry
- it is compatible with a modular approach to design and enjoys a close relationship to a familiar supporting *T5* topology – the hierarchical Systems Engineering framework common in the automotive industry.

However at the *T1* level, the topology tends to be expensive in terms of sensors, communications and control hardware. More importantly though, there are alternative topologies which enable a much greater degree of flexibility and fault tolerance.

3.3 Decentralised and Heterarchical Control

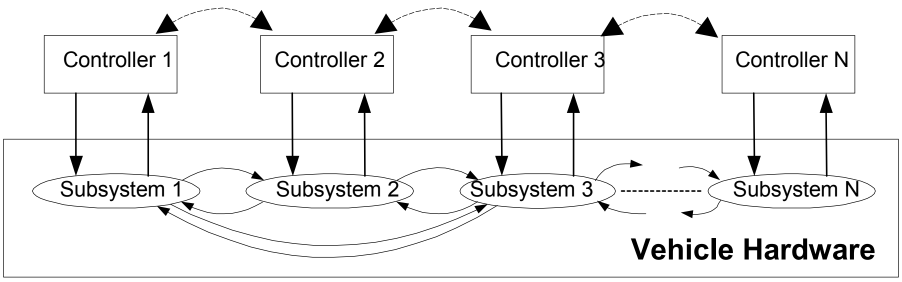

Fig. 7. Decentralised and heterarchical control

Decentralised control (Fig. 7) has an obvious parallel structure and is essentially the basic non-integrated case of vehicle controllers (designed and) implemented for each subsystem independently, and with no inter-system communication. Instead of adding supervisory action, 'integration' may be achieved by the addition of communications between the controllers, allowing each to interact with some or all of the others; this gives rise to the so-called *heterarchical* architecture sometimes considered in the flexible control of manufacturing systems - e.g.[43]. The obvious lack of global structure would seem to make this approach wholly unsuitable for safety critical control in road vehicles, and it does not seem to have been considered in this context at all. Of course it might be that a suitably constrained algorithmic structure would make this *T2* topology feasible in the future, but for all of its obvious flexibility, the lack of an operational control structure makes it unlikely to be treated seriously.

4. MULTI-LAYER ARCHITECTURES

Here we consider a general layered architecture which is naturally abstracted from the hierarchical two and three-layer control (*T2*) topologies of Section 3. This leads to a single class of functional (*T5*) topologies, but which admits alternative structures in terms of the flow of control actions (*T2*).

Fig. 8 shows a number of levels or layers of vehicle control functionality or ‘behaviours’. Higher layers involve more integrated functionality, in other words a more coordinated use of actuators. The bottom layer, Layer 0, represents the minimum level of functionality for the vehicle to operate at all, whether or not any electronic control systems (e.g. drive-by-wire) are included. Because, for simplicity, the sensor and information structures are not being dealt with in any detail, the interface to such systems is simply represented on the left side of the figure in terms of a multiplexed ‘bus’ or networked server from which information is interchanged. There are many possibilities for how information is structured and exchanged, for example with basic sensor signals being supplemented by state and disturbance estimates from dedicated information processing systems, or via the control blocks shown, or both.

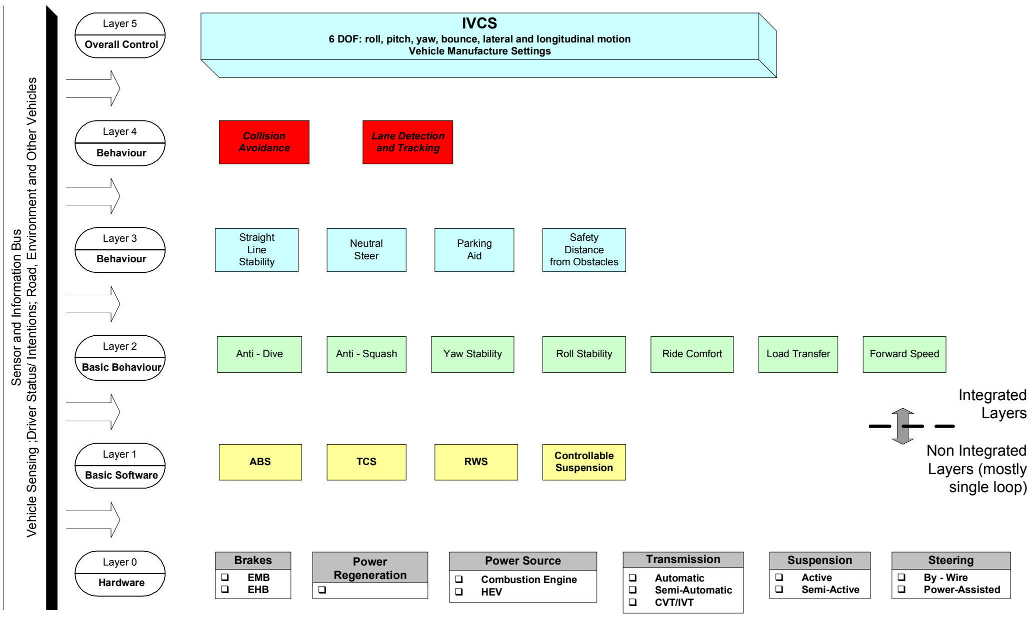

Fig. 8. Multi-layered vehicle control design architecture

The blocks shown are not proposed as a definitive list of vehicle control functions, nor as a IVCS standard; rather the figure illustrates the kind of increasing levels of ‘integration’ in the higher layers that might be implemented in practice, and it would be for each vehicle manufacturer (or systems integrator) to define and develop such functionality in any particular case. It is of course quite possible to extend the layering upwards to include multi-vehicle system behaviour and interaction with other roadway systems; e.g. in [44] such extensions are considered within a layered architecture, albeit in the aerospace context.

Each block represents an individual ‘behaviour’ [45,46] that is to be integrated into the overall control system operation. While in principle a ‘behaviour’ could be any type of operational activity that has no obvious overall function (e.g. increase driver braking demand signal), in any realistic application in IVCS, designed behaviours would all have some desirable function. For example in Layer 1 'ABS' could represent the limited function of avoiding excessive longitudinal slip within braking a single wheel by releasing or holding brake pressure at the wheel cylinder, whilst 'Yaw Stability' in Layer 2 might represent the use of brakes to control excessive oversteer. And it becomes a matter of design philosophy whether the control of understeer via braking is treated as a separate behaviour, and similarly the use of drive-torque distribution for yaw control might be treated as a separate behaviour.

This figure does not include any vertical arrows to define the flow of control commands, and there are very many ways to achieve this. So Fig. 8 represents a multiplicity of control topologies that potentially share a common layered structure, rather than a single prescribed architecture. The essential feature is the system modularity, and hence design flexibility – each block represents a 'sub-system controller' that may be designed separately to achieve desired functional requirements.

Three associated control topologies are now considered. It is important to recognise that these cases have implications not only for the connectivity and protocols that determine the overall control authority (and perhaps the available classes of control algorithm) but also the design methodologies that underpin them; these have significant implications for broader aspects such as flexibility discussed above.

4.1 Hierarchical Control

In this top-down approach, the required vehicle level behaviour is initially defined, broken down into sub-tasks, sub-sub tasks etc., in a traditional 'Systems Engineering' cascade. Higher layers are responsible for the overall goals and objectives of the integrated vehicle system, while lower layers are responsible for solving the resulting sub-problems, e.g. by tracking reference input signals. A serial structure is typical [6, 34,35,47,48] with communication only possible between adjacent layers, as in the simplified Fig. 9.

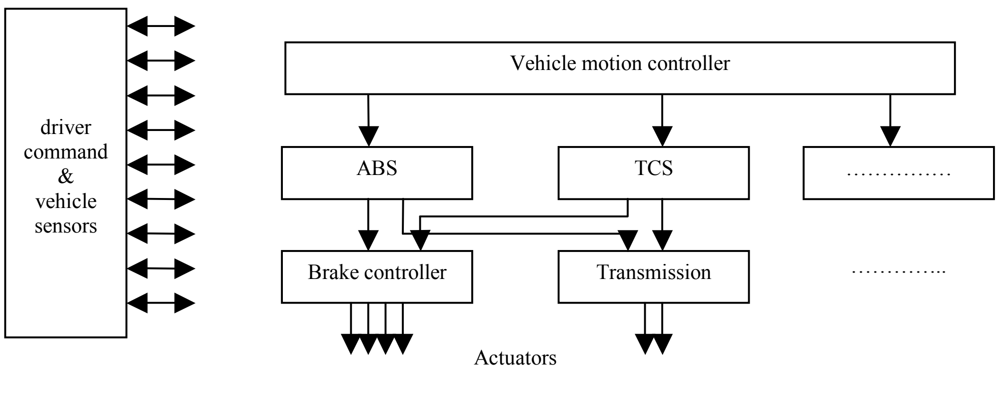

Fig. 9. Hierarchical Integrated Vehicle Control System

The figure illustrates a feasible *T2*-topology, with the *T5* structure still 'visible' in the layering. The flow of control actions is top-down, with higher layers directly acting as inputs – typically as reference signals – to lower layers. Strictly, all modules must be operational for the IVCS to function, so if the physical (*T1*) topology is similar or identical to the control topology, hardware redundancy becomes be necessary for all of the modules shown. As has already been discussed, in a pure form the architecture is inherently inflexible from the design perspective – for example the addition of a new actuator potentially leads to the redesign of the entire IVCS.

4.2 Subsumption Architecture

The subsumption architecture [45,46] is derived via a bottom-up design methodology. As originally proposed, behaviours (control functional objectives) operate in parallel within any layer, and without higher level supervision. Higher layers intermittently take over or 'subsume' control from lower layers, and hence there would be no single coordinating vehicle level controller – different control functions, across all layers operate asynchronously and in parallel. The layering simultaneously determines authority over control actions (higher layers dominate), design sequencing (lower layers first) and fault response (regress to lower level competency). The fundamental property of this architecture is that higher layers subsume the actions of lower ones by applying a bias to, or completely suppressing, lower level control commands. Lower levels continues to function as higher levels are added, 'unaware' of any interference with their activity. A feasible flow of commands is represented in Fig. 10, where the circular nodes represent a switching function: when the higher layer activates a control signal it predominates, replacing that of any lower layer(s).

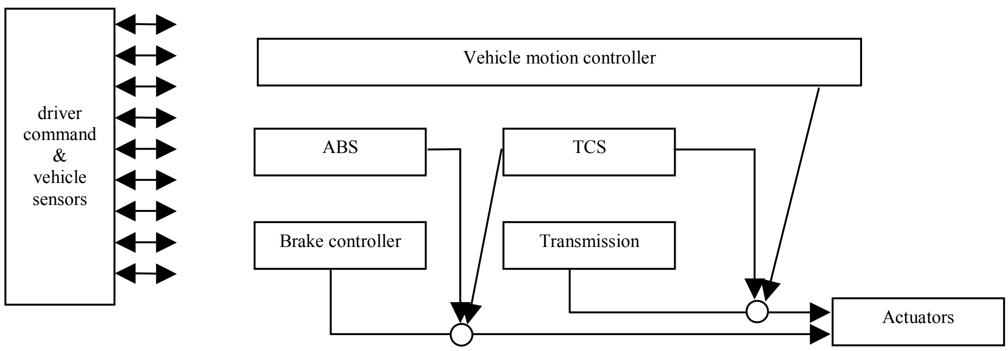

Fig. 10. Subsumption based control topology

The introduction of additional switching nodes, plus relaxing the strict hierarchy of layer-to-layer communication greatly increases the number of ways that a series of control modules or behaviours can be assembled, and justifiably raises questions of increased complexity and unpredictable outcomes. On the other hand, the idea that just the basic Layer 0 functionality is sufficient for the vehicle to operate, and that both design and implementation of higher layers is incremental, offers distinct advantages in terms of flexibility and fault tolerance; for example, only the lowest layers absolutely require hardware redundancy for fault tolerant operation; when higher level functions are compromised by faults or failures, the idea would be to shut down the behaviours immediately affected.

Although formally introduced in the context of layered behaviour-based control, the above approach is already familiar in the context of standard ABS control systems – see for example [49,50]. In practice an ABS system may intermittently override a driver demand for increasing braking torque whenever there is danger of locking the wheels, by either releasing or holding pressure at the wheel cylinders. The natural scope for fault detection and management of failure modes is also implemented in practice – once a fault is detected, the ABS is simply disengaged and a warning communicated to the driver [49].

4.3 Hybrid Architectures

The architectures shown above are in ‘pure form’ and can be easily generalised and hybridised. For example, the connectivity in Fig. 11 combines both hierarchical and subsumption features. In the vehicle context it seems natural to retain a subsumption type of control at lower levels, whilst introducing a more hierarchical approach within the highest layers.

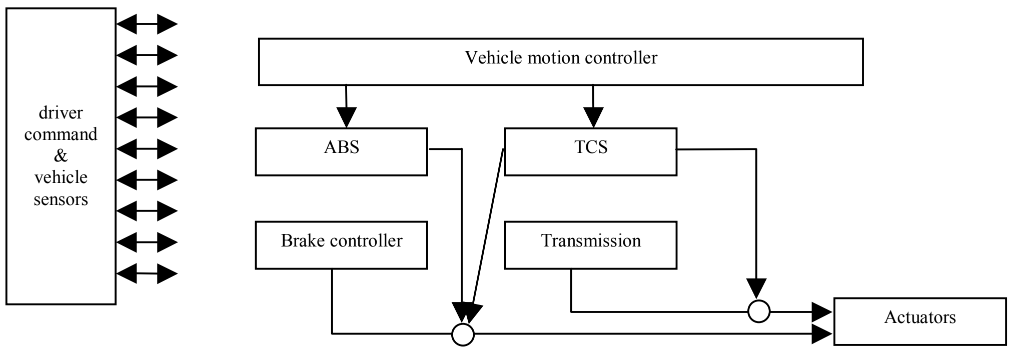

Fig. 11. Hybrid topology

An ambiguous feature can be seen in Figs. 10 and 11 – where multiple arrows enter a single node, and the subsuming commands arise from the same layer, it is uncertain which control predominates. And even without this problem, is it always the case that higher layers ‘know best’ in subsumption? While it might be argued that higher level controls will only occasionally interrupt, and the lower level functionality is not severely disturbed, this ‘interventionist’ approach is clearly not absolutely necessary. To sidestep further discussion, it is obvious that the ‘subsumption nodes’ shown above are special cases of a situation where multiple systems are potentially competing for the use of a smaller number of actuators, and some form of coordination mechanism is needed. Subsumption (highest rank wins) is only one of many possibilities.

Before considering further coordination mechanisms, two essential features should be noted in the above: (i) all actuator signals are uniquely sourced from a coordination node or a control block – there is no ambiguity in the actuator input, (ii) there is no expectation that coordinators have 'greater knowledge' of the overall vehicle control task than the subsystem controllers feeding actuation demands. Indeed to require this would make the coordinator blocks more like 'total system controllers', and hence regress from a flexible modular control design methodology to a complex and highly redundant variant of the centralised/hierarchical topology.

5. COORDINATED CONTROL

This section considers how state-of-the-art coordinated control can be applied to vehicle systems. The focus now shifts away from architectures (some form of multi-layer design architecture will now be assumed) and control algorithms (any number might be used for sub-system controller design); emphasis is instead on the coordination mechanisms that 'glue together' the overall IVCS.

Fig. 12 depicts four basic coordinator types, which are representative of the methods described in the literature of behaviour-based control (e.g. [33,46]). In Fig. 12(a) the coordination block is one of 'pure subsumption' as introduced in the previous section; the highest ranked (highest level) non-zero command is transmitted to the output. This is an example of competitive 'conflict resolution' where a single behaviour dominates and takes control of an actuator. Fig. 12(b) represents another competitive approach where the largest (modulus) activation signal is chosen – this approach is essentially where the coordinator 'listens to the loudest voice'.

Both (a) and (b) clearly involve step changes in control, as the coordinator switches between modes, and there is a clear danger that this will create undesirable transients in the vehicle dynamic response. Of course it is possible to low-pass filter the output, or include a ramped transition within the coordinator rather than an instantaneous switch.

Cooperative coordination occurs when some kind of superposition or averaging is carried out. This could be in the form of simple averaging, or via a non-linear interpolation function provided by an Artificial Neural Network (Fig. 12c). In either case, the linear or non-linear function weights need to be chosen to 'optimise' the coordinated control performance in some sense. The potential advantage of using a nonlinear interpolation function is that actuator saturation may be avoided, and that smooth transitions can be made between different modes of coordination as input signal amplitudes change. It is also quite feasible to include additional state or sensor information as input into the neural network; this would allow for example a greater emphasis on safe and stable performance when the vehicle is moving at high speed. However, if taken to extremes, a neural network coordinator could itself become a complex non-linear multivariable controller, overly-dependent on the detailed control activity of the supplying sub-system controllers.

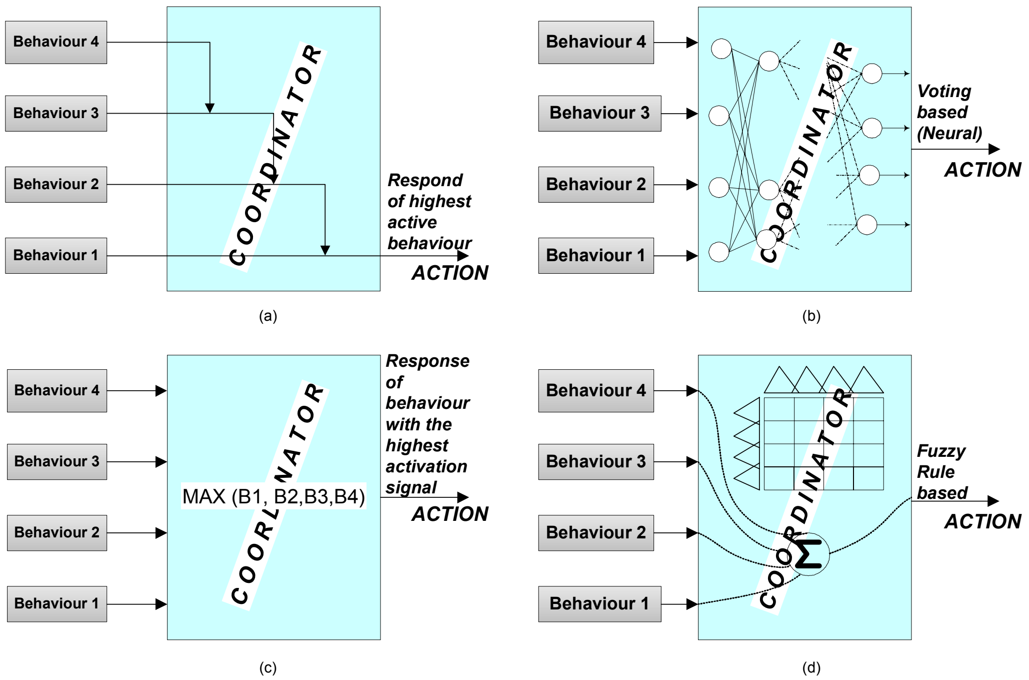

Fig. 12. Examples of control action coordination

Fig. 12(d) represents a fuzzy rule-based coordination function, which though also inherently nonlinear, has the advantage that the underlying rule-base may be easily understood in everyday terms. For example a 'fuzzy subsumption' coordinator may use membership functions for 'small' and 'large' input signals in place of the 'crisp' 'off' and 'on' states for normal subsumption. The same logical decisions are used (highest level predominates), but the fuzzy system again provides smooth transitions.

Fuzzy coordination is also very natural in the case where input sub-system controllers are themselves fuzzy systems; in this case the inputs may remain fuzzy variables, in the form of aggregated fuzzy sets; the coordinator can then performs additional aggregation, and defuzzification is only applied at the final step before actuation [51].

The use of fuzzy control and (crisp) subsumption is now demonstrated in a simple example of integrated vehicle control. Three layers are evident in Fig. 13, which are now described in turn. Note that the coordinator marked 'F' uses fuzzy aggregation of fuzzy inputs, while those marked 'S' use standard subsumption with the filled arrow (⟶▶) predominating. Also note that in this case the 'lower' ABS/TCS behaviour subsumes control from the 'higher behaviour' of Yaw Stability; this is strictly against the philosophy of the subsumption approach to behaviour-based control, where higher layers normally predominate. (Here there is clearly no point locking wheels to attempt to improve yaw stability, when this would overrides the ABS controller and actually tend to reduce lateral stability!)

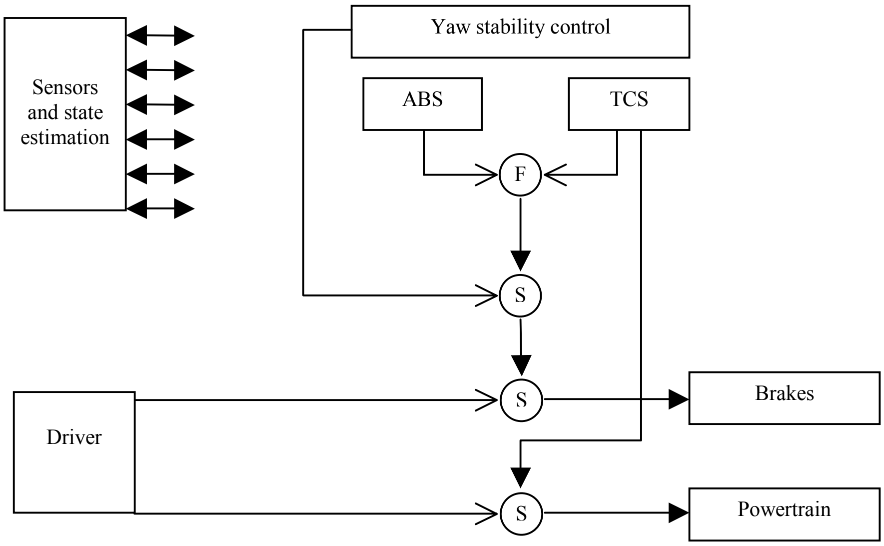

Fig. 13. Fuzzy coordinated control of vehicle dynamics

Layer 0 – Basic hardware
This covers the basic driver-to-actuator control of steering, driving torque and brakes, which for simulation comprises a vehicle simulation model and a driver model. For simplicity, the vehicle model has three major degrees of freedom – forward, lateral and yaw - and a 'stiff suspension' that eliminates body bounce, roll and pitch, but allows lateral and longitudinal load transfer, and thereby the essential tyre nonlinearities. There are also four dynamic wheel-spin degrees of freedom and nonlinear tyre characteristics modelled using a combined slip Pacejka formula. As well as the dynamic degrees of freedom, the model includes simple controls and transients for wheel braking, driveline torque and front-wheel steer. The vehicle model is described further in reference [52], as is the combined longitudinal/lateral driver model. The model is based on a simple combination of a 'reference vector field' feedforward policy, plus proportional-integral (PI) feedback control. The driver model operates throughout the whole linear and nonlinear range of the vehicle dynamics, but is deliberately 'unsophisticated' – for example the PI tracking of reference yaw rate works on the assumption that 'more steering implies more yaw velocity' which may well not be the case, for example with limit understeer behaviour. Full details of the driver model are given in [52].

While it is possible to criticise the integrated modelling of driver and vehicle to assess closed-loop dynamic behaviour, on the grounds that the results depend on the specific details of the driver model, its use is becoming increasingly widespread (e.g.

[20,22,53,54]) and provided the driver model is sufficiently general and robust, the resulting trends (as described below) are most likely to be meaningful.

Layer 1 – Non-integrated ABS/TCS control

The ABS controller uses a 5×5 matrix of control rules, based on a computer error between wheel slip λ and a desired (reference) value. The control output is defined, based on the error and the (discrete-time) change in error. As is common in fuzzy inference systems, triangular membership functions for input and output are associated with fuzzy sets such as 'positive big' (PB) etc.; for full detail see the Appendix, Table A1. The ABS controller is applied to each wheel independently, as would be expected for the low level non-integrated control.

The TCS controller is similarly based on fuzzy inference (Table A2) except that both brakes and driveline are controlled, and a logic table is required for each output. Also, since both ABS and TCS potentially 'compete' for the use of the brakes, a fuzzy aggregation coordinator is required, as in Fig. 12 (even though ABS and TCS generally operate under very different driving conditions).

Layer 2 – Integrated yaw stability control

This is a fuzzy/sliding mode controller, the error signal being determined by deviations from a sliding surface, and a rule-base in constructed to reduce this error and hence ensure stable control – see Appendix. It is designed only to correct *oversteering* vehicle behaviour, and in line with the strategy being demonstrated here, if *understeer* corrections were required, an additional controller could be defined, and fuzzy aggregation applied to coordinate the two sets of demands; for simplicity this is not included in the current example. The yaw controller operates on the brakes, and only when vehicle yaw rate exceeds a certain threshold. A combination of brake actuators is selected in order to generate a moment that acts to reduce the instantaneous vehicle yaw velocity.

The following results demonstrate some expected performance benefits of including the two-layer integrated control, compared to relying on driver skill and basic vehicle hardware. In both cases the driver attempts to negotiate an 'S-bend' with the vehicle entry speed 'too high' for stable vehicle motion along the desired path, the centre-line of the track. Simulation starts at the entry to the first bend, with a vehicle speed of 30 ms^{-1}; the driver recognises the problem (very late) and attempts to brake towards a reference speed of 20 ms^{-1}, reducing to 17 ms^{-1} through the curves (sufficiently slow for stable centre-line path following) then increasing to a new target speed of 25 ms^{-1} on the second straight section. Fig. 14 shows the vehicle paths, and Fig. 15 presents a number of dynamic response variables (dotted line is without integrated control in both figures).

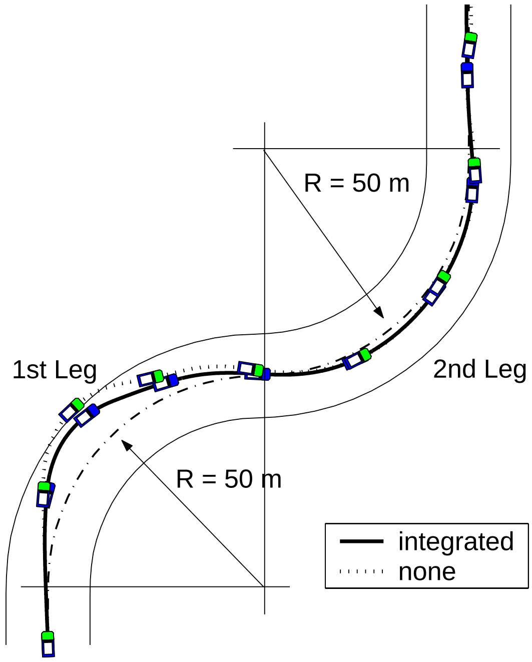

Fig. 14. Vehicle Path – severe braking through an S-bend

The vehicle paths are not too dissimilar, though it is clear that the vehicle without integrated control veers off the track on the first bend. In both cases the vehicle manages to return close to the centreline on the second bend, but the basic vehicle is much harder to control, as can be seen in Fig. 15 where a number of dynamic responses are displayed. In the steering plot, it is clear to see that the driver's work load is reduced when integrated vehicle control is used, especially in the second half of the simulation.

Vehicle lateral stability is most clearly represented in the plots of vehicle side-slip angle and yaw velocity; compared to the basic vehicle, these variables are very well controlled, even under aggressive driving. The basic (rear-wheel drive) vehicle is hard to control by the driver when maximum acceleration is demanded in the second half of the simulation; indeed an oversteer condition leads to a yaw oscillation under closed-loop control, which grows until the driver completely loses stable control of the vehicle later in the simulation.

There is almost no oscillation on the forward speed as the longitudinal slip is controlled to be the optimal one for the specific surface. Considering the acceleration, the vehicle with integrated control allows slightly less and constant forward acceleration than the vehicle without integrated control, keeping its stability.

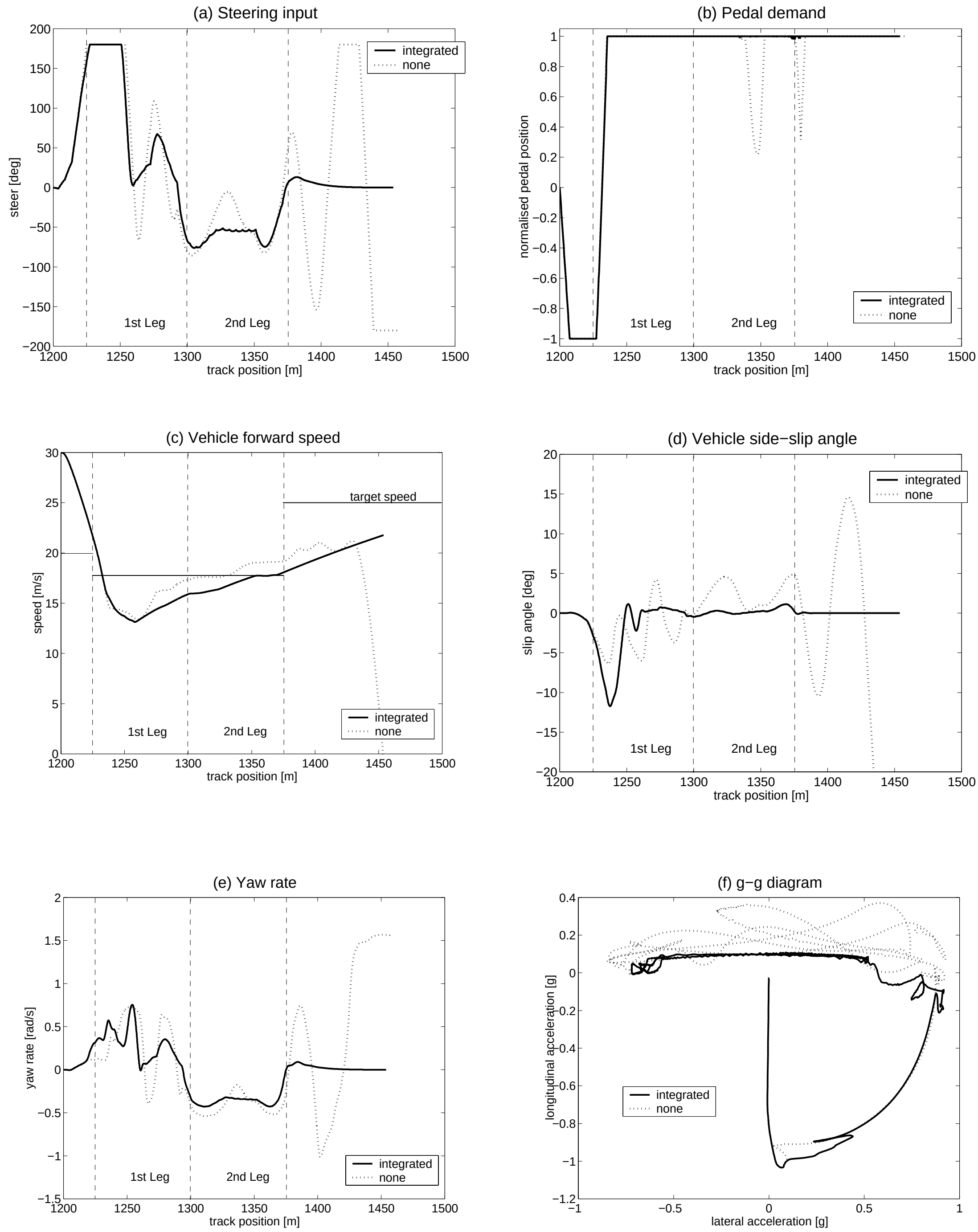

Fig. 15. Dynamic response effect of the integrated vehicle control system

The plot in Fig. 15(f) is a g-g diagram, which shows a clear and expected advantage of the integrated control (compare Fig. 2). In particular it is possible to simultaneously brake and steer in a controlled fashion near the limits of friction. It is worth noting that the apparently greater longitudinal accelerations possible for the basic vehicle are merely due to poor vehicle side-slip angle control; in this case, when the vehicle rotates towards the cornering direction, lateral accelerations attain an additional component that appears to be longitudinal in the vehicle-fixed axis system. In any case, these excursions of acceleration in the forward direction are clearly very badly controlled, and coincide with loss of stability on the second straight section of track.

The IVCS presented here is quite rudimentary; for example there are *no* circumstances under which wheel lockup is allowed; subsumption provides a permanent bias towards the basic ABS/TCS functionality. However, in the presence of large slip angles (and in the absence of an active steer degree of freedom) the direction of the force vector at any wheel can only be properly controlled by inducing such lockup or wheel-spin.

However the example serves to demonstrate that useful integrated vehicle control can be implemented by adhering to a multi-layer 'bottom up' design architecture, implemented by a relatively simple coordination policy applied to independently designed sub-system controllers.

6. TOWARDS DYNAMIC COORDINATION IN IVCS

The advantages of the above general approach to integrated vehicle control is clear: it provides a modular and incremental design method, providing adequate scope for separating out local and integrated design responsibilities. Within vehicle development, its use would rely heavily on simulation – so the modular approach would mean sub-system suppliers needing to provide validated dynamic model of the closed-loop controller/hardware, to allow the IVCS to be developed in simulation. This latter development would include the detailed choice of *T2*/*T3* architectures, the types of coordinator and the tuning of these coordinators (e.g. via choice input weights).

One problem with the approach is the very large amount of freedom still available for integration. If an inappropriate type of coordination is chosen for example, the overall system may exhibit poor performance, or even generate dynamic instabilities in certain circumstances. One particular issue is that the coordination function is relatively decoupled from the performance objectives of the subsystem controllers, providing what might be seen as an undesirable freedom to degrade the overall control. Hence the purpose of this section is to explore further possibilities (ones that do not so far appear to have been the subject of detailed research).

The essential features of a coordinator are that (i) it improves the 'integrated dynamic performance' of the vehicle and (ii) it makes decisions based directly on information obtained from the sub-system controllers (e.g. control signal amplitude). This includes the possibility of requiring a 'richer' form of control information from sub-system controllers than the desired actuator commands. In fact this is already the case with coordination via fuzzy aggregation, where the fuzzy inputs contain significantly more information than the (crisp) control demands.

Where sub-system controllers are based on formal optimal control, an enriched control signal may comprise - in addition to the actuator demands - local sensitivity information in the form of an 'internal sub-system Hamiltonian function' [10]. If the Hamiltonian function from the optimal control is computed in the subsystem, as a function of control output, then this immediately provides a dynamic sensitivity function for the additional cost of any sub-optimal control. Summing the input sensitivity functions ϕ_1 and ϕ_2, and minimising the result provides a 'dynamically optimised' control signal $\bar{u}(t)$; the coordinator formally recognises and balances the control objectives of the sub-system controllers. Fig. 16 illustrates this situation where a scalar control is coordinated between two sub-systems. In reference [10] the sub-systems (simulated active suspension corner actuators) were controlled via linear optimal control (LQR), and the minimisation was carried out off-line: an explicit closed-form integrated controller was obtained, and performance was similar to an equivalent 'global' centralised controller derived from a full vehicle dynamic model. However, the on-line minimisation proposed here has the major advantage that the coordinator needs no specific information of the sub-system states and models. In the case of linear optimal control theory, the sensitivity functions are quadratic, and even in the multivariable case the solution is closed-form, requiring only one matrix inversion.

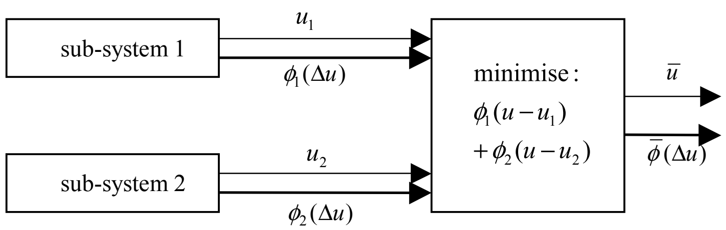

Fig. 16. Hamiltonian (cost sensitivity) coordinator concept

In the figure, an output sensitivity function is also shown, as this would be necessary if the coordinator output were to be fed into a second Hamiltonian coordinator. Clearly weighting parameters can also be included to vary the relative importance of the sub-system demands. Although it will not be discussed further here, one major advantage of this approach is the *potential* to develop overall stability criteria for the IVCS. This might be developed from the Lyapunov functions directly available from the sub-system controllers. In any case, provided the sub-system control criteria are compatible in some sense, this dynamically based control integration should always work to improve the overall system performance compared to non-integrated control, promoting safety, stability and confidence in the resulting IVCS.

In the above, two types of dynamic coordination have been suggested – a weak form based on fuzzy aggregation, and a more fundamentally based approach based on formal (linear) optimal control theory. To be really feasible in practice, such coordination needs to be applicable to a wider range of subsystem control methods. The key requirement is that there exists a form of error criterion, where minimising this criterion automatically means achieving the desired performance outcome of the sub-system controller. One form of control for which this is applicable is sliding mode control, where deviation from the sliding surface can be used as a measure of error. And more generally, where nonlinear or adaptive control makes explicit use of a Lyapunov function, the dynamic change in this function can also be developed as a measure of error for coordinator operation. The fundamental development required for control implementation is that such criteria need to be explicitly available in real-time, rather than simply implicit within the controller design.

Perhaps the most common type of low-level dynamic control currently employed on vehicles (and elsewhere in industrial control systems) is single-loop PID control. There is unlikely to exist any universal formula for costing perturbations in such controllers; the ‘dynamic cost sensitivity’ is likely to be highly dependent on the application. However it is easy to conceive of a ‘retrofit’ sensitivity analysis, even if this is not carried out within the original design. Controls are perturbed in system simulation and performance objectives assessed over time in response to these perturbations. Modelling of the performance dependency may then be undertaken via a variety of techniques (system identification, neural networks, polynomial response surfaces, etc.).

It has already been pointed out that the layered design architectures offer possibilities for robust and fault-tolerant vehicle control systems, based on the simple idea that if faults develop in hardware or software that is specific to higher layers, these can be shut down and the lower layers continue to function. So only the ‘basic hardware’ needs to incorporate expensive component redundancy. But fault tolerant behaviour may also be derived from the ‘intelligent’ use of coordination functions. This has been suggested in [10] where a simple actuator malfunction is seen to be compensated for by (on-line) optimisation that takes account of diagnostic information on the failure. The major advantage is that this dynamic fault compensation requires no *a-priori* knowledge of anticipated failure modes.

7. CONCLUSIONS

This paper has reviewed examples from the literature on Integrated Vehicle Control Systems (particularly chassis control systems) and attempted to discern the main structural features of such control, and establish where there is scope for new developments, not just theoretically, but also taking account of the practical commercial constraints in vehicle development. Related literature from robotics and aerospace vehicles has also been taken into account.

One simple but important conclusion is that control system architecture has several levels of meaning, and in this paper five fairly distinct levels have been discerned. This means that, with the possible exception of fully decentralised or fully centralised concepts, it is insufficient to formulate *the* control architecture in terms of a single topology. A related conclusion is the integrated control is a very much richer concept than multivariable control, with requirements that it should:-

1. be modular - so that design may be distributed between teams, and for example there is no need to redesign existing systems when additional sub-systems, sensors or actuators are added
2. respect the intellectual property of suppliers - sub-system controllers need to collaborate, without the need to make public all algorithmic details
3. avoid excessive complexity (e.g. as measured by real-time processing, communication and memory requirements) – hence to grow 'reasonably' with the number of actuators and sensors on the vehicle
4. incorporate fault detection, diagnosis and tolerance
5. possess an open architecture that permits equal status 'access' to additional systems, both internal and external.

The most promising candidate for the required structured approach is through layered design, similar to that employed in 'behaviour-based' control. The term 'sub-system' controller has been used interchangeably with the 'control behaviour', both essentially referring to the delivery of a specific form of dynamic performance capability, rather than some specific use of an actuator or a subset of states. In this approach the design of *control coordinators* becomes the major task for vehicle control integration, and some specific types have been explored. One such method has been demonstrated, where a combination of fuzzy sub-system controllers, with coordination via fuzzy aggregation and (crisp) subsumption has been seen to develop integrated control behaviour at the vehicle level.

The excessive design freedom of this approach has motivated the need for seeking closer links between sub-system controllers and the associated control coordinators, and one potential approach has been suggested.

There is one key aspect of IVCS not covered in this paper, and this concerns the role of the driver and his or her connection to the control architecture. In the multi-layered architectures, the driver may clearly interact simultaneously will all layers of the integrated control system. This follows from the sequential design methodology of adding competency with each new layer, and is quite different from typical hierarchical approaches where the driver is to interact only at the highest level in the hierarchy (e.g. [6,34,47]).

The final conclusion is that there still remains a significant research challenge to properly understand and implement IVCS's in future vehicles, even without considering the greater challenges of understanding how such advanced systems should best interact with the driver, external vehicles and highways systems.

REFERENCES

1. Anon.: Daimler-Chrysler HITECH REPORT, Research and Technology Issue 2 (2002), p.67.
2. Freuchte, R.D., Karmel, A.M. and Rillings, J.H.: Integrated vehicle control. IEEE Vehicular Technology Conference, 2 (1989), pp.868-877.
3. Tanaka, H., Inoue, H. and Iwata, H.: Development of a vehicle integrated control system. Proc. of the XXIV FISITA Congress, June 1992, London. (1992), pp.63-72.
4. Wallentowitz, H.: Integration of chassis and traction control systems. what is possible - what makes sense - what is under development. AVEC 92 (1992), pp.1-7.
5. Kiencke, U.: Integrated vehicle control systems. IFAC: Intelligent Components for Autonomous and Semi-Autonomous Vehicle, Toulouse, France (1995), pp.1-5.
6. Coelingh, E., Chaumette, P. and Andersson, M.: Open-interface definitions for automotive systems. SAE 2002 World Congress (2002) 2002-01-0267.
7. Hirano, Y., Harada, H., Ono, E. and Takanami, K.: Development of an integrated system of 4WS and 4WD by H-infinity control. SAE Journal 930267 (1993), pp.79-86.
8. Busshardt, J., Chaumette, P., Hac, A., Hoang-Gia, S., Hoying, J., Oakley, R. and O'Connell S.: An integrated chassis control system to enhance vehicle stability. Progress in Technology. 77 (1999), pp.373-382
9. Ghoneim, Y.A., Lin, W.C., Sidlosky, D.M., Chen, H.H., Chin, Y.K. and Tedrake, M.J.: Integrated chassis control system to enhance vehicle stability. International Journal of Vehicle Design, 23 (2000), pp.124-144.
10. Gordon, T.J..: An integrated strategy for the control of a full vehicle active suspension system. Vehicle System Dynamics, 25 (1996), pp.229-242.
11. Wills, L., Kannan, S., Sander, S., Murat Guler, M., Heck, B., Prasad, J., Schrage, D., and Vachtsevanos, G.: An open platform for reconfigurable control. IEEE Control Systems Magazine, 21(3), (2001), pp.49-64.
12. Shladover, S.: Review of the state of development of advanced vehicle control systems (AVCS). Vehicle System Dynamics vol.24 (1995), pp.551-595.
13. Abu-Lebdeh, G.: Integrated adaptive-signal dynamic-speed control of signalized arterials. Journal of Transportation Engineering, 128 (2002), pp. 447-451.
14. Sato, S., Inoue, H., Tabata, M. and Inagaki, S.: Integrated chassis control system for improved vehicle dynamics. AVEC 92 (1992), pp.413-418
15. Nagai, M., Yamanaka, S. and Hirano, Y.: Integrated control of active rear wheel steering and yaw moment control using braking forces. JSME International Journal, 42 (2) Series C (1999), pp.301-308.
16. Wang, Y.Q. and Nagai M.: Improvement of vehicle handling and stability by an integrated four-wheel-steer and yaw moment control system. Progress in Technology, 77 (1999), pp.173-182.
17. You, S.S. and Chai, Y.H.: Multi-objective control synthesis: an application to 4WS passenger vehicles. Mechatronics (1999), pp.363-390.
18. Horiuchi, S., Okada, K. and Nohtomi, S.: Effects of integrated control of active wheel steering and individual wheel torque on vehicle handling and stability - a comparison of alternative control strategies. Vehicle System Dynamics Supplement 33 (1999), pp.680-691.
19. Mastinu, G., Babbel E., Lugner P., Margolis D., Mittermayr, P. and Richter, B.: Integrated controls of lateral vehicle dynamics. Vehicle System Dynamics (Supplement) 23 (1994), pp.358-377.
20. Rodic, A.D. and Vukobratovic, M.K.: Contribution to the integrated control synthesis of road vehicles. IEEE Transactions on Control Systems Technology, 7 (1999), pp.64-78.

21. Tai, M. and Tomizuka, M.: Robust lateral control of heavy duty vehicles for automated highway systems. Proc. 1999 14th IFAC Triennial World Congress, Beijing, (1999), pp.37-42.
22. Mokhiamar, O. and Abe, M.: Active wheel steering and yaw moment control combination to maximize stability as well as vehicle responsiveness during quick lane change for active vehicle handling safety. Proc. Inst. Mech. Eng. 216 Part D (2002), pp.115-124.
23. Lim, E.H.M. and Hedrick, J.K.: Lateral and longitudinal vehicle control coupling for automated vehicle operation. American Control Conference (1999), pp.3676-3680.
24. Komatsu, A., Gordon, T.J. and Best, M.C.: 4WS control of handling dynamics using a linear optimal reference model. Proceedings of the 5th International Symposium on Advanced Vehicle Control (AVEC2000) , Ann Arbor, USA, (2000), pp.253-260.
25. Economou, J.T. and Colyer R.E.: Modelling of skid steering and fuzzy logic vehicle ground interaction. American Control Conference (2000), pp.100-104.
26. Lee, C.: Fuzzy logic in control systems : fuzzy logic controller -Part I and Part II. IEEE Trans. Syst. Man Cybern. 20 (1990), pp.404-434.
27. Zeyada, Y.F., ElBeheiry, E.M. and ElArabi, M.E.: Fault diagnosis and mu-estimation for integrated chassis control of vehicles in emergency maneuvers. Journal of Engineering and Applied Science, Cairo, 47 (2000), pp.665-682.
28. Rodic, A.D. and Vukobratovie, M.K.: Design of an integrated active control system for road vehicles operating with automated highway systems. International Journal of Computer Application in Technology, 13 (2000), pp.78-92.
29. Nwagboso, C.O., Ouyang, X. and Morgan, C.: Development of neural-network control of steer-by-wire system for intelligent vehicles. Heavy Vehicle Systems, 9 (2002), pp.1-26.
30. Katebi, M.R. and Grimble, M.J.: Integrated control, guidance and diagnosis for reconfigurable autonomous underwater vehicle control. International Journal of Systems Science, 30 (1999), pp.1021-1032.
31. Harada, M. and Harada, H.: Analysis of lateral stability with integrated control of suspension and steering systems. JSAE 20 (1999), pp.465-470.
32. Evans, J., Hodge, W., Liebman, J., Tomlin, C. and Parkinson, B.: Flight tests of an unmanned air vehicle with integrated multi-antenna GPS receiver and IMU; towards a testbed for distributed control and formation flight. Proceedings of the ION-GPS Conference, Nashville, (1999).
33. Carreras, M., Batlle, J., Ridao, P. and Roberts, G.N.: An overview on behaviour-based methods for AUV control. 5th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC2000). Aalborg, Denmark (2000), pp.141-146.
34. Palkovics, L.A., and Fries, A.: Intelligent electronic systems in commercial vehicles for enhanced traffic safety. Vehicle System Dynamics, 35 (2001), pp.227-289.
35. Phillips, A.M.: Functional decomposition in a vehicle control system. American Control Conference (2002), pp.3713-3718.
36. Schmidt, E.H., Wright, C.D. and Zwerne, J.S.: Required elements of integrated vehicle control systems. Proc. International Congress on Transportation Electronics, SAE paper 901170 (1990), pp.463-471.
37. Rago, C., Prasanth, R., Mehra, R.K. and Fortenbaugh, R.: Failure detection and identification and fault tolerant control using the Imm-Kf with applications to the eagle-eye UAV. Decision & Control Tampa, Florida, USA (1998), pp.4208-4213.
38. Bartolini, A., Sheppard, J.W. and Munns, T.E.: An application of diagnostic inference modelling to vehicle health management. IEEE Systems Readiness Technology Conference Piscataway, NJ, USA (2001), pp.706-717.
39. Nagai, M. Hirano, Y. and Yamanaka, S.: Integrated robust control of active rear wheel steering and direct yaw moment control. Vehicle System Dynamics (Supplement) 28 (1998), pp.416-421.
40. Ono, E., Takanami, K., Iwama, N., Hayashi, Y., Hirano, Y. and Satoh, Y.: Vehicle integrated control for steering and traction systems by mu-synthesis. Automatica 30 (1994), pp.1639-1647.

41. Brennan, S. and Alleyne, A.: Integrated vehicle control via coordinated steering and wheel torque inputs. American Control Conference (2001), pp.7-12.
42. Kelling, N.A., and Heck, W.: The BRAKE project – centralized versus distributed redundancy for brake-by-wire systems. SAE 2002 World Congress (2002) 2002-01-0266.
43. Duffie, N.A., Chitturi, R. and Mou.,J.: Fault-tolerant heterarchical control of heterogeneous manufacturing system entities. Journal of Manufacturing Systems, 7 (1988), pp.315-328.
44. Keller, K., Wiegand, D., Swearingen, K., Reisig, C., Black, S., Gillis, A. and Vandernoot, M.: An architecture to implement integrated vehicle health management systems. IEEE Systems Readiness Technology Conference, Piscataway, USA, (2001), pp.2-15.
45. Brooks, R. A.: A Robust Layered Control System for a Mobile Robot. IEEEJournal of Robotics and Automation, 2(1), (1986), pp. 14-23
46. Brooks, R.A.: Integrated systems based on behaviors. SIGART Bulletin 2:4, August (1991), pp. 46-50.
47. Girard, A.R., De Sousa, J.B., Misener, J.A. and Hedrick, J.K.: A control architecture for integrated cooperative cruise control and collision warning systems. IEEE Conference on Decision and Control (2001), pp.1491-1496.
48. Smakman, H.: Functional integration of active suspension with slip control for improved lateral vehicle dynamics. AVEC 2000 Ann Arbor, Michigan, USA (2000), pp.397-404.
49. Austin, L. and Morreu, D.: Recent advances in antilock braking systems and traction control systems. Proc. Inst. Mech. Eng., Part D Journal of Automotive Engineering, 214 (2000), pp.625-639.
50. Bosch.: Driving Safety-Systems, Robert Bosch GmbH, 2nd Ed., SAE, Warrendale, USA, 1999.
51. Hagras, H., Callaghan, V., Colley, M. and Carr-West, M.: A behaviour based hierarchical fuzzy control architecture for agricultural autonomous mobile robots. The International Conference on Computational Intelligence for Modelling, Control and Automation, Vienna-Austria (1999), pp.172-177.
52. Gordon, T.J., Best, M.C. and Dixon, P.J.: An automated driver based on convergent vector fields. Proc. Inst. of Mech. Eng., Part D Automobile Engineering , 216D4 (2002), pp.329-347.
53. Beji, L. and Bestaoui, Y.: An adaptive control method of automated vehicles with integrated longitudinal and lateral dynamics in road following, robot motion and control. (RoMoCo'01) Bukowy Dworek, Poland (2001), pp.201-206.
54. Becker, U., Rodic, A. and Schnieder, E.: Integrated modeling of driver-vehicle dynamics for use in designing of driver-assisted control systems. Automatisierungs- und Assistenzsysteme fur Transportmittel, Brunswick, Germany (2002), pp.108-129.
55. Buckholtz, K.R.: Use of fuzzy logic in wheel slip assignment - part I: yaw rate control. SAE Journal (2002), 2002-01-1221.

APPENDIX

The simulation study contained in Section 5 is based on the vehicle and driver models described in [52]; the (*T2*) control topology is described in Section 5, together with the of form of the subsumption coordinators, so this Appendix is provided to summarise the fuzzy sub-system controllers.

The ABS controller is a fuzzy inference system based on the 25 rules given below in Table A1, together with standard triangular membership functions. The TCS is also a fuzzy controller with 25 rules for braking actuation (Table A2) on the driven wheels and another 25 rules for the control of driving torque (Table A3). The TCS controller also makes use of standard triangular membership functions.

Table A1. ABS control rules

ABS		error				
		PB	PS	ZR	NS	NB
Change of error	PB	NB	NB	NB	NS	PB
	PS	NB	NB	NS	NS	PB
	ZR	NB	NB	ZR	ZR	PB
	NS	NB	NS	ZR	PS	PB
	NB	NB	NS	ZR	PB	PB

Table A2. TCS control rules (brake control)

TCS-Brake		error				
		PB	PS	ZR	NS	NB
Change of error	PB	PB	PB	ZR	ZR	ZR
	PS	PB	PS	ZR	ZR	ZR
	ZR	PB	ZR	ZR	ZR	ZR
	NS	PS	ZR	ZR	ZR	ZR
	NB	PS	ZR	ZR	ZR	ZR

Table A3. TCS control rules (drive torque control)

TCS-Drive		error				
		PB	PS	ZR	NS	NB
Change of error	PB	NB	NB	ZR	ZR	ZR
	PS	NB	NB	ZR	ZR	ZR
	ZR	NB	NS	ZR	ZR	ZR
	NS	NB	NS	ZR	ZR	ZR
	NB	NB	NS	ZR	ZR	ZR

Table A4. Yaw stability control rules

YAW	Outer-Front	Outer-Rear	Inner-Front	Inner-Rear
N4	ZR	P4	ZR	P4
N3	ZR	P4	ZR	P3
N2	ZR	P4	ZR	P2
N1	ZR	ZR	ZR	P1
ZR	ZR	ZR	ZR	ZR
P1	P1	ZR	ZR	ZR
P2	P2	ZR	P4	ZR
P3	P3	ZR	P4	ZR
P4	P4	ZR	P4	ZR

For these two longitudinal slip controllers, the error is defined as

$$e = \lambda - \lambda_{ref}$$

where λ is the actual or estimated longitudinal wheel slip and λ_{ref} is the target reference value.

The yaw stability controller is a combined sliding mode-fuzzy controller based on that presented in reference [55]. The error and change of error signals are combined in a sliding surface defined by the equation

$$s = e + \gamma \dot{e}$$

where $e(t)$ is a tracking error in yaw velocity r :

$$e = r - r_{ref}$$

and the reference yaw rate is given in terms of lateral acceleration and forward speed:

$$r_{ref} = \frac{a_y}{v_x}.$$

The sliding mode signal $s(t)$ is presented to the fuzzy logic controller, which commands distributed wheel braking, designed to reduce the error towards zero. The output signals from the fuzzy controller are labelled Outer-Front, Outer-Rear, etc., where 'Outer' and 'Inner' are defined relative to the cornering direction of the the roadway (or intended vehicle path).

The fuzzy control has 9 rules with 4 outputs for each rule – Table A4. The input membership function are equally distributed across the fuzzy input 'universe' (-1,1). *N4* is a *Z* function, *P4* is an *S* function, all the others are standard Gaussian functions. The output membership function are all Gaussian functions with universe is defined from 0 to 1, and 1 corresponds to maximum braking.